IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently Amended) A high performance network address processor comprising:

a longest prefix match lookup table for receiving a network address request having a designated network destination address, the longest prefix match lookup table having multiple pipelined lookup tables, a first pipelined lookup table having a single row of a first set of data pairs, the first set of data pairs including a key value and a mask value, the mask value indicating a number of least significant bits that may be ignored within the key value, the first set of pairs being ordered according to corresponding mask values, a second pipelined lookup table having a plurality of rows of a second set of data pairs; and

an associated data engine coupled to the longest prefix match lookup table that is capable of receiving a key and an output address pointer from the longest prefix match lookup table and that is capable of providing a network address processor data output corresponding to the designated network address pointer.

2. (Currently amended) The high performance network address processor of claim 1 wherein the longest prefix match lookup engine includes a third pipelined lookup table having a plurality of rows of a third set of data ~~pairs~~ tuples, the data tuples including a key value, a mask value and a pointer value.

3. (Previously presented) The high performance network address processor of claim 1 wherein a value representing a position of a selected element in the single row of a first set of data pairs is an input to the second pipelined lookup table, the input used to select one of the plurality of rows of the second set of data pairs.

4. (Previously presented) The high performance network address processor of claim 2 wherein a value representing a position of a selected element in the single row of a first set of data pairs is a first input to the second pipelined lookup table, the input used to select one of the plurality of rows of the second set of data pairs, a second input into the second pipelined lookup table is used to locate one of the data pairs of the one of the plurality of rows.

5. (Currently Amended) A high performance network address processor integrated circuit, wherein the network address processor integrated circuit comprises:

a longest prefix match lookup table engine for receiving a network address request having a designated network destination address, the longest prefix match lookup table engine having a plurality of pipelined lookup tables, a first pipelined lookup table having a single row of a first set of data pairs, the first set of data pairs including a key value and a mask value, the mask value indicating a number of least significant bits that may be ignored within the key value, the first set of data pairs being ordered according to corresponding mask values, and a second pipelined lookup table having a plurality of rows of a second set of data pairs; and

an associated data engine coupled to the longest prefix match lookup table that is capable of receiving a key value and an output address pointer from the longest prefix match lookup table and that is capable of providing a network address processor data output corresponding to the designated network address pointer, the associated data engine having a first lookup table having a plurality of rows, wherein a portion of bits of the key value is used to select one of the plurality of rows as an output, a remaining portion of the bits of the key value identifying a row in a second lookup table having a plurality of rows.

6. (Currently amended) The high performance network address processor of claim 5 wherein the longest prefix match lookup engine a third pipelined lookup table having a plurality of rows of a third set of data tuples, ~~pairs~~ the data tuples including a key value, a mask value and a pointer value.

7. (Previously presented) The high performance network address processor of claim 5 wherein a value representing a position of a selected element in the single row of a first set of data pairs is an input to the second pipelined lookup table, the input used to select one of the plurality of rows of the second set of data pairs.

8. (Currently Amended) A high performance network addressing method comprising:

providing a longest prefix match lookup engine with a network address data request and a destination network address, wherein the longest prefix match lookup engine comprises a set of lookup tables, each of the set of lookup tables including data pairs including a key

value and a mask, the mask value indicating a number of least significant bits that may be ignored within the key value, the data pairs being ordered in each lookup table according to corresponding masks;

searching the set of lookup tables to select a look up engine address output from the set of lookup tables, the successive searching including,

selecting a position within a row of a first lookup table;

identifying a value associated with the position;

utilizing the value as a first input to a second lookup table;

selecting a row of the second lookup table according to the first input;

selecting a position within the row of the second lookup table according to a second input; and

accessing a value stored in the position within the row of the second lookup table;

defining a pointer to be provided ~~provide~~ as input to an associated data engine; and

searching the associated data engine to provide an associated destination address output.

9. (Currently Amended) The high performance network addressing method of claim 8 wherein successively searching the set of lookup tables comprises the smallest entry that is greater than or equal to an input search key, includes,

selecting the smallest entry that equals the input search key with a corresponding number of mask bits,

wherein if one or more entries comprise a same key, a key having a smallest mask is selected, and

wherein if no key matches, a maximum key in a row is compared with the input search key using each set of respective mask pointer pairs, each of the pointer pairs is selected to correspond to the smallest mask for which the input search key equals the maximum key in a row of a corresponding lookup table with the corresponding number of mask bits ignored.

10. (new) The high performance network address processor of claim 1 wherein the first set of data pairs are ordered such that a pair having a lowest mask value is at an end of the single row.

11. (new) The high performance network address processor of claim 5 wherein the first set of data pairs are ordered such that a pair having a lowest mask value is at an end of the single row.